ZigBee/IEEE 802.15.4 Summary

Sinem Coleri Ergen
Email: csinem@eecs.berkeley.edu

September 10, 2004

Abstract

This document gives the motivation for the ZigBee alliance and explains the physical, medium access and routing layers of ZigBee.

Contents

Chapter 1

Introduction

1.1 Evolution of LR-WPAN Standardization

The cellular network was a natural extension of the wired telephony network that became pervasive during the mid-20th century. As the need for mobility and the cost of laying new wires increased, the motivation for a personal connection independent of location to that network also increased. Coverage of large area is provided through (1-2km) cells that cooperate with their neighbors to create a seemingly seamless network. Examples of standards are GSM, IS-136, IS-95. Cellular standards basically aimed at facilitating voice communications throughout a metropolitan area.

During the mid-1980s, it turned out that an even smaller coverage area is needed for higher user densities and the emergent data traffic. The IEEE 802.11 working group for WLANs is formed to create a wireless local area network standard.

Whereas IEEE 802.11 was concerned with features such as Ethernet matching speed, long-range(100m), complexity to handle seamless roaming, message forwarding, and data throughput of 2-11Mbps, WPANs are focused on a space around a person or object that typically extends up to 10m in all directions. The focus of WPANs is low-cost, low power, short range and very small size. The IEEE 802.15 working group is formed to create WPAN standard. This group has currently defined three classes of WPANs that are differentiated by data rate, battery drain and quality of service(QoS). The high data rate WPAN(IEEE 802.15.3) is suitable for multi-media applications that require very high QoS. Medium rate WPANs (IEEE 802.15.1/Blueetooth) will handle a variety of tasks ranging from cell phones to PDA communications and have QoS suitable for voice communications. The low rate WPANs(IEEE 802.15.4/LR-WPAN) is intended to serve a set of industrial, residential and medical applications with very low power consumption and cost requirement not considered by the above WPANs and with relaxed needs for data rate and QoS. The low data rate enables the LR-WPAN to consume very little power.

1.2 ZigBee and IEEE 802.15.4

ZigBee technology is a low data rate, low power consumption, low cost, wireless networking protocol targeted towards automation and remote control applications. IEEE 802.15.4 committee started working on a low data rate standard a short while later. Then the ZigBee Alliance and the IEEE decided to join forces and ZigBee is the commercial name for this technology.

ZigBee is expected to provide low cost and low power connectivity for equipment that needs battery life as long as several months to several years but does not require data transfer rates as high as those enabled by Bluetooth. In addition, ZigBee can be implemented in mesh networks larger

than is possible with Bluetooth. ZigBee compliant wireless devices are expected to transmit 10-75 meters, depending on the RF environment and the power output consumption required for a given application, and will operate in the unlicensed RF worldwide(2.4GHz global, 915MHz Americas or 868 MHz Europe). The data rate is 250kbps at 2.4GHz, 40kbps at 915MHz and 20kbps at 868MHz.

IEEE and ZigBee Alliance have been working closely to specify the entire protocol stack. IEEE 802.15.4 focuses on the specification of the lower two layers of the protocol(physical and data link layer). On the other hand, ZigBee Alliance aims to provide the upper layers of the protocol stack (from network to the application layer) for interoperable data networking, security services and a range of wireless home and building control solutions, provide interoperability compliance testing, marketing of the standard, advanced engineering for the evolution of the standard. This will assure consumers to buy products from different manufacturers with confidence that the products will work together.

IEEE 802.15.4 is now detailing the specification of PHY and MAC by offering building blocks for different types of networking known as **"star, mesh, and cluster tree"**. Network routing schemes are designed to ensure power conservation, and low latency through **guaranteed time slots**. A unique feature of ZigBee network layer is **communication redundancy** eliminating "single point of failure" in mesh networks. Key features of PHY include energy and link quality detection, clear channel assessment for **improved coexistence with other wireless networks**.

1.3 ZigBee vs. Bluetooth

ZigBee looks rather like Bluetooth but is simpler, has a lower data rate and spends most of its time snoozing. This characteristic means that a node on a ZigBee network should be able to run for six months to two years on just two AA batteries. (HOW?)

The operational range of ZigBee is 10-75m compared to 10m for Bluetooth(without a power amplifier).

ZigBee sits below Bluetooth in terms of data rate. The data rate of ZigBee is 250kbps at 2.4GHz, 40kbps at 915MHz and 20kbps at 868MHz whereas that of Bluetooth is 1Mbps.

ZigBee uses a basic master-slave configuration suited to static star networks of many infrequently used devices that talk via small data packets. It allows up to 254 nodes. Bluetooth's protocol is more complex since it is geared towards handling voice, images and file transfers in ad hoc networks. Bluetooth devices can support scatternets of multiple smaller non-synchronized networks(piconets). It only allows up to 8 slave nodes in a basic master-slave piconet set-up.

When ZigBee node is powered down, it can wake up and get a packet in around 15 msec whereas a Bluetooth device would take around 3sec to wake up and respond.

Chapter 2

IEEE 802.15.4 WPAN

The main features of this standard are network flexibility, low cost, very low power consumption, and low data rate in an adhoc self-organizing network among inexpensive fixed, portable and moving devices. It is developed for applications with relaxed throughput requirements which cannot handle the power consumption of heavy protocol stacks.

2.1 Components of WPAN

A ZigBee system consists of several components. The most basic is the device. A device can be a full-function device (FFD) or reduced-function device (RFD). A network shall include at least one FFD, operating as the PAN coordinator.

The FFD can operate in three modes: a personal area network (PAN) coordinator, a coordinator or a device. An RFD is intended for applications that are extremely simple and do not need to send large amounts of data. An FFD can talk to RFDs or FFDs while an RFD can only talk to an FFD.

2.2 Network Topologies

Figure 2.1 shows 3 types of topologies that ZigBee supports: star topology, peer-to-peer topology and cluster tree.

2.2.1 Star Topology

In the star topology, the communication is established between devices and a single central controller, called the PAN coordinator. The PAN coordinator may be mains powered while the devices will most likely be battery powered. Applications that benefit from this topology include home automation, personal computer (PC) peripherals, toys and games.

After an FFD is activated for the first time, it may establish its own network and become the PAN coordinator. Each start network chooses a PAN identifier, which is not currently used by any other network within the radio sphere of influence. This allows each star network to operate independently.

2.2.2 Peer-to-peer Topology

In peer-to-peer topology, there is also one PAN coordinator. In contrast to star topology, any device can communicate with any other device as long as they are in range of one another. A peer-to-peer

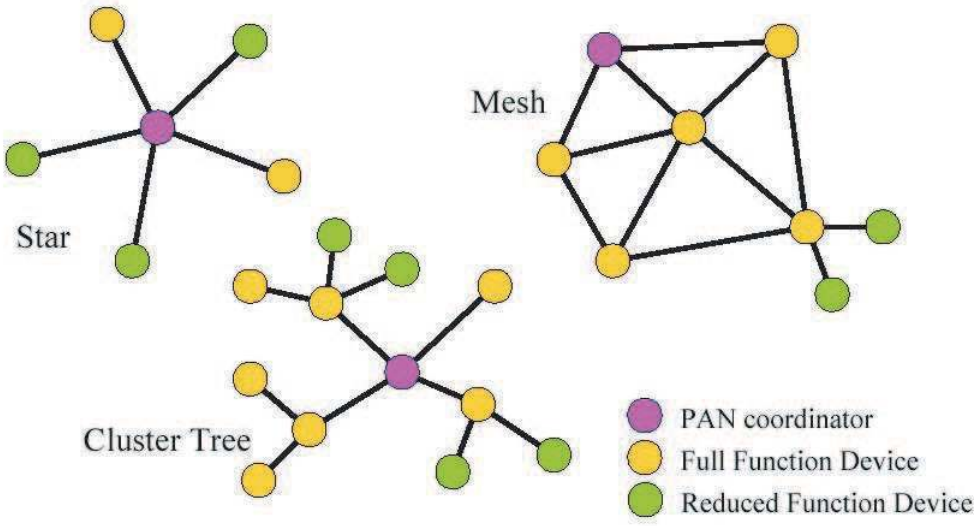

Figure 2.1: Topology Models.

network can be ad hoc, self-organizing and self-healing. Applications such as industrial control and monitoring, wireless sensor networks, asset and inventory tracking would benefit from such a topology. It also allows multiple hops to route messages from any device to any other device in the network. It can provide reliability by multipath routing.

2.2.3 Cluster-tree Topology

Cluster-tree network is a special case of a peer-to-peer network in which most devices are FFDs and an RFD may connect to a cluster-tree network as a leave node at the end of a branch. Any of the FFD can act as a coordinator and provide synchronization services to other devices and coordinators. Only one of these coordinators however is the PAN coordinator.

The PAN coordinator forms the first cluster by establishing itself as the cluster head (CLH) with a cluster identifier (CID) of zero, choosing an unused PAN identifier, and broadcasting beacon frames to neighboring devices. A candidate device receiving a beacon frame may request to join the network at the CLH. If the PAN coordinator permits the device to join, it will add this new device as a child device in its neighbor list. The newly joined device will add the CLH as its parent in its neighbor list and begin transmitting periodic beacons such that other candidate devices may then join the network at that device. Once application or network requirements are met, the PAN coordinator may instruct a device to become the CLH of a new cluster adjacent to the first one. The advantage of this clustered structure is the increased coverage area at the cost of increased message latency.

2.3 LR-WPAN Device Architecture

Figure 2.2 shows an LR-WPAN device. The device comprises a PHY, which contains the radio frequency (RF) transceiver along with its low-level control mechanism, and a MAC sublayer that provides access to the physical channel for all types of transfer. The upper layers consists of a network layer, which provides network configuration, manipulation, and message routing, and application layer, which provides the intended function of a device. An IEEE 802.2 logical link control

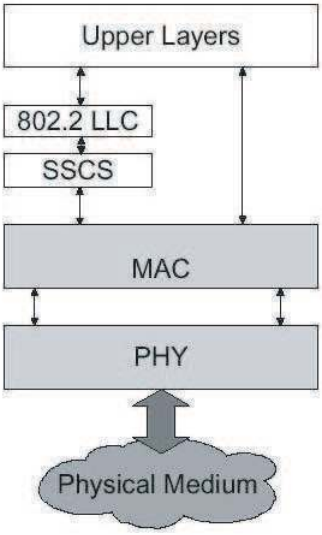

Figure 2.2: LR-WPAN Device Architecture.

(LLC) can access the MAC sublayer through the service specific convergence sublayer (SSCS).

Chapter 3 describes the physical layer of IEEE 802.15.4. Chapter 4 explains the MAC layer of IEEE 802.15.4. Chapter 5 gives the routing mechanisms that are going to be used in the ZigBee.

Chapter 3

IEEE 802.15.4 PHY

The PHY provides two services: the PHY data service and PHY management service interfacing to the physical layer management entity (PLME). The PHY data service enables the transmission and reception of PHY protocol data units (PPDU) across the physical radio channel.

The features of the PHY are activation and deactivation of the radio transceiver, energy detection (ED), link quality indication (LQI), channel selection, clear channel assessment (CCA) and transmitting as well as receiving packets across the physical medium.

The standard offers two PHY options based on the frequency band. Both are based on direct sequence spread spectrum (DSSS). The data rate is 250kbps at 2.4GHz, 40kbps at 915MHz and 20kbps at 868MHz. The higher data rate at 2.4GHz is attributed to a higher-order modulation scheme. Lower frequency provide longer range due to lower propagation losses. Low rate can be translated into better sensitivity and larger coverage area. Higher rate means higher throughput, lower latency or lower duty cycle. This information is summarized in Figure 3.1.

PHY (MHz)	Frequency band (MHz)	Spreading parameters		Data parameters		
		Chip rate (kchip/s)	Modulation	Bit rate (kb/s)	Symbol rate (ksymbol/s)	Symbols
868/915	868–868.6	300	BPSK	20	20	Binary
	902–928	600	BPSK	40	40	Binary
2450	2400–2483.5	2000	O-QPSK	250	62.5	16-ary Orthogonal

Figure 3.1: Frequency bands and data rates.

There is a single channel between 868 and 868.6MHz, 10 channels between 902.0 and 928.0MHz, and 16 channels between 2.4 and 2.4835GHz as shown in Figure 3.2. Several channels in different frequency bands enables the ability to relocate within spectrum. The standard also allows dynamic channel selection, a scan function that steps through a list of supported channels in search of beacon, receiver energy detection, link quality indication, channel switching.

Receiver sensitivities are -85dBm for 2.4GHz and -92dBm for 868/915MHz. The advantage of 6-8dB comes from the advantage of lower rate. THE ACHIEVABLE RANGE IS A FUNCTION OF RECEIVER SENSITIVITY AND TRANSMIT POWER.

The maximum transmit power shall conform with local regulations. A compliant device shall have its nominal transmit power level indicated by the PHY parameter, *phyTransmitPower*.

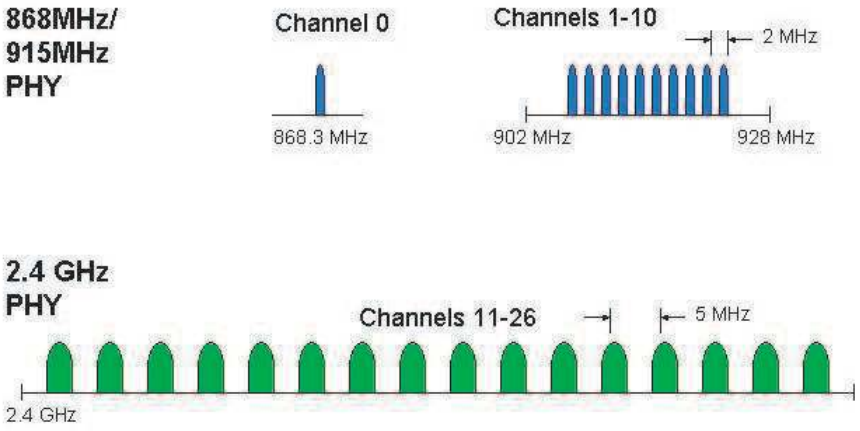

Figure 3.2: Operating frequency bands.

3.1 Receiver Energy Detection (ED)

The receiver energy detection (ED) measurement is intended for use by a network layer as part of channel selection algorithm. It is an estimate of the received signal power within the bandwidth of an IEEE 802.15.4 channel. No attempt is made to identify or decode signals on the channel. The ED time should be equal to 8 symbol periods.

The ED result shall be reported as an 8-bit integer ranging from $0x00$ to $0xff$. The minimum ED value (0) shall indicate received power less than 10dB above the specified receiver sensitivity. The range of received power spanned by the ED values shall be at least 40dB. Within this range, the mapping from the received power in decibels to ED values shall be linear with an accuracy of $+-6dB$.

3.2 Link Quality Indication (LQI)

Upon reception of a packet, the PHY sends the PSDU length, PSDU itself and link quality (LQ) in the PD-DATA.indication primitive. The LQI measurement is a characterization of the strength and/or quality of a received packet. The measurement may be implemented using receiver ED, a signal-to-noise estimation or a combination of these methods. The use of LQI result is up to the network or application layers.

The LQI result should be reported as an integer ranging from $0x00$ to $0xff$. The minimum and maximum LQI values should be associated with the lowest and highest quality IEEE 802.15.4 signals detectable by the receiver and LQ values should be uniformly distributed between these two limits.

3.3 Clear Channel Assessment (CCA)

The clear channel assessment (CCA) is performed according to at least one of the following three methods:

Octets: 4	1	1		variable
Preamble	SFD	Frame length (7 bits)	Reserved (1 bit)	PSDU
SHR		PHR		PHY payload

Figure 3.3: Format of the PPDU.

- Energy above threshold. CCA shall report a busy medium upon detecting any energy above the ED threshold.
- Carrier sense only. CCA shall report a busy medium only upon the detection of a signal with the modulation and spreading characteristics of IEEE 802.15.4. This signal may be above or below the ED threshold.
- Carrier sense with energy above threshold. CCA shall report a busy medium only upon the detection of a signal with the modulation and spreading characteristics of IEEE 802.15.4 with energy above the ED threshold.

3.4 PPDU Format

The PPDU packet structure is illustrated in Figure 3.3. Each PPDU packet consists of the following basic components:

- SHR, which allows a receiving device to synchronize and lock into the bit stream
- PHR, which contains frame length information
- a variable length payload, which carries the MAC sublayer frame.

Chapter 4

IEEE 802.15.4 MAC

The MAC sublayer provides two services: the MAC data service and the MAC management service interfacing to the MAC sublayer management entity (MLME) service access point (SAP) (MLME-SAP). The MAC data service enables the transmission and reception of MAC protocol data units (MPDU) across the PHY data service.

The features of MAC sublayer are beacon management, channel access, GTS management, frame validation, acknowledged frame delivery, association and disassociation.

4.1 Superframe Structure

LR-WPAN allows the optional use of a superframe structure. The format of the superframe is defined by the coordinator. The superframe is bounded by network beacons and is divided into 16 equally sized slots. The beacon frame is sent in the first slot of each superframe. If a coordinator does not want to use the superframe structure, it may turn off the beacon transmissions. The beacons are used to synchronize the attached devices, to identify the PAN and to describe the structure of superframes.

The superframe can have an active and an inactive portion. During the inactive portion, the coordinator shall not interact with its PAN and may enter a low-power mode. The active portion portion consists of contention access period (CAP) and contention free period (CFP). Any device wishing to communicate during the CAP shall compete with other devices using a slotted CSMA-CA mechanism. On the other hand, the CFP contains guaranteed time slots (GTSs). The GTSs always appear at the end of the active superframe starting at a slot boundary immediately following the CAP. The PAN coordinator may allocate up to seven of these GTSs and a GTS can occupy more than one slot period.

The duration of different portions of the superframe are described by the values of *macBeaconOrder* and *macSuperFrameOrder*. *macBeaconOrder* describes the interval at which the coordinator shall transmit its beacon frames. The beacon interval, BI, is related to the *macBeaconOrder*, BO, as follows: $BI = aBaseSuperFrameDuration 2^{BO}, 0 \leq BO \leq 14$. The superframe is ignored if $BO = 15$.

The value of *macSuperFrameOrder* describes the length of the active portion of the superframe. The superframe duration, SD, is related to *macSuperFrameOrder*, SO, as follows: $SD = aBaseSuperFrameDuration 2^{SO}, 0 \leq SO \leq 14$. If $SO = 15$, the superframe should not remain active after the beacon.

The active portion of each superframe is divided into a *aNumSuperFrameSlots* equally spaced slots of duration $2^{SO} aBaseSlotDuration$ and is composed of three parts: a beacon, a CAP and

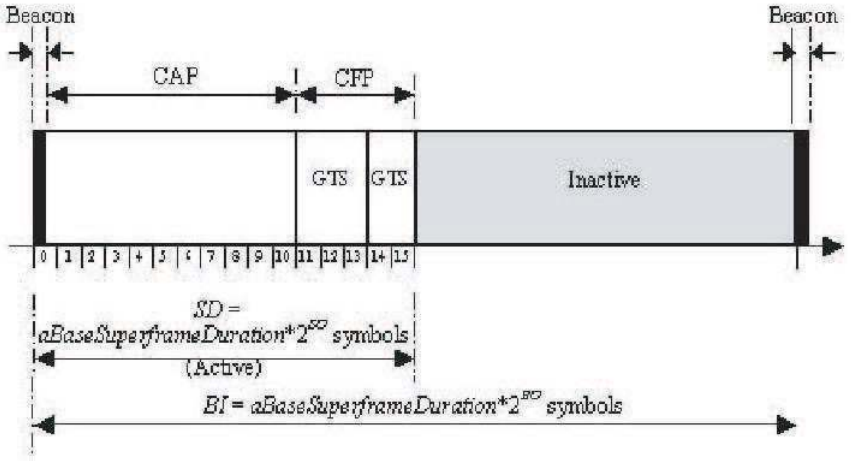

Figure 4.1: An example superframe structure.

CFP. The beacon is transmitted at the start of slot 0 without the use of CSMA. The CAP starts immediately after the beacon. The CAP shall be at least *aMinCAPLength* symbols unless additional space is needed to temporarily accommodate the increase in the beacon frame length to perform GTS maintenance. All frames except acknowledgement frames or any data frame that immediately follows the acknowledgement of a data request command that are transmitted in the CAP shall use slotted CSMA-CA to access the channel. A transmission in the CAP shall be complete one IFS period before the end of the CAP. If this is not possible, it defers its transmission until the CAP of the following superframe. An example superframe structure is shown in Figure 4.1.

The CFP, if present, shall start on a slot boundary immediately following the CAP and extends to the end of the active portion of the superframe. The length of the CFP is determined by the total length of all of the combined GTSs. No transmissions within the CFP shall use a CSMA-CA mechanism. A device transmitting in the CFP shall ensure that its transmissions are complete one IFS period before the end of its GTS.

IFS time is the amount of time necessary to process the received packet by the PHY. Transmitted frames shall be followed by an IFS period. The length of IFS depends on the size of the frame that has just been transmitted. Frames of up to *aMaxSIFSFrameSize* in length shall be followed by a SIFS whereas frames of greater length shall be followed by a LIFS.

The PANs that do not wish to use the superframe in a nonbeacon-enabled shall set both *macBeaconOrder* and *macSuperFrameOrder* to 15. In this kind of network, a coordinator shall not transmit any beacons, all transmissions except the acknowledgement frame shall use unslotted CSMA-CA to access channel, GTSs shall not be permitted.

4.2 CSMA-CA Algorithm

If superframe structure is used in the PAN, then slotted CSMA-CA shall be used. If beacons are not being used in the PAN or a beacon cannot be located in a beacon-enabled network, unslotted CSMA-CA algorithm is used. In both cases, the algorithm is implemented using units of time called backoff periods, which is equal to *aUnitBackoffPeriod* symbols.

In slotted CSMA-CA channel access mechanism, the backoff period boundaries of every device in the PAN are aligned with the superframe slot boundaries of the PAN coordinator. In slotted CSMA-CA, each time a device wishes to transmit data frames during the CAP, it shall locate the

boundary of the next backoff period. In unslotted CSMA-CA, the backoff periods of one device do not need to be synchronized to the backoff periods of another device.

Each device has 3 variables: NB, CW and BE. NB is the number of times the CSMA-CA algorithm was required to backoff while attempting the current transmission. It is initialized to 0 before every new transmission. CW is the contention window length, which defines the number of backoff periods that need to be clear of activity before the transmission can start. It is initialized to 2 before each transmission attempt and reset to 2 each time the channel is assessed to be busy. CW is only used for slotted CSMA-CA. BE is the backoff exponent, which is related to how many backoff periods a device shall wait before attempting to assess the channel. Although the receiver of the device is enabled during the channel assessment portion of this algorithm, the device shall discard any frames received during this time.

In slotted CSMA-CA, NB, CW and BE are initialized and the boundary of the next backoff period is located. In unslotted CSMA-CA, NB and BE are initialized (step1). The MAC layer shall delay for a random number of complete backoff periods in the range 0 to $2^{BE} - 1$ (step 2) then request that PHY performs a CCA (clear channel assessment) (step 3). The MAC sublayer shall then proceed if the remaining CSMA-CA algorithm steps, the frame transmission, and any acknowledgement can be completed before the end of the CAP. If the MAC sublayer cannot proceed, it shall wait until the start of the CAP in the next superframe and repeat the evaluation.

If the channel is assessed to be busy (step 4), the MAC sublayer shall increment both NB and BE by one, ensuring that BE shall be no more than *aMaxBE*. In slotted CSMA-CA, CW can also be reset to 2. If the value of NB is less than or equal to *macMaxCSMABackoffs*, the CSMA-CA shall return to step 2, else the CSMA-CA shall terminate with a Channel Access Failure status.

If the channel is assessed to be idle (step 5), in a slotted CSMA-CA, the MAC sublayer shall ensure that contention window is expired before starting transmission. For this, the MAC sublayer first decrements CW by one. If CW is not equal to 0, go to step 3 else start transmission on the boundary of the next backoff period. In the unslotted CSMA-CA, the MAC sublayer start transmission immediately if the channel is assessed to be idle. The whole CSMA-CA algorithm is illustrated in Figure 4.2.

4.3 Data Transfer model

Three types of data transfer transactions exist: from a coordinator to a device, from a device to a coordinator and between two peer devices. The mechanism for each of these transfers depend on whether the network supports the transmission of beacons.

When a device wishes to transfer data in a nonbeacon-enabled network, it simply transmits its data frame, using the unslotted CSMA-CA, to the coordinator. There is also an optional acknowledgement at the end as shown in Figure 4.3.

When a device wishes to transfer data to a coordinator in a beacon-enabled network, it first listens for the network beacon. When the beacon is found, it synchronizes to the superframe structure. At the right time, it transmits its data frame, using slotted CSMA-CA, to the coordinator. There is an optional acknowledgement at the end as shown in Figure 4.4.

The applications transfers are completely controlled by the devices on a PAN rather than by the coordinator. This provides the energy-conservation feature of the ZigBee network. When a coordinator wishes to transfer data to a device in a beacon-enabled network, it indicates in the network beacon that the data message is pending. The device *periodically listens to the network beacon*, and if a message is pending, transmits a MAC command requesting this data, using slotted CSMA-CA. The coordinator optionally acknowledges the successful transmission of this packet. The pending

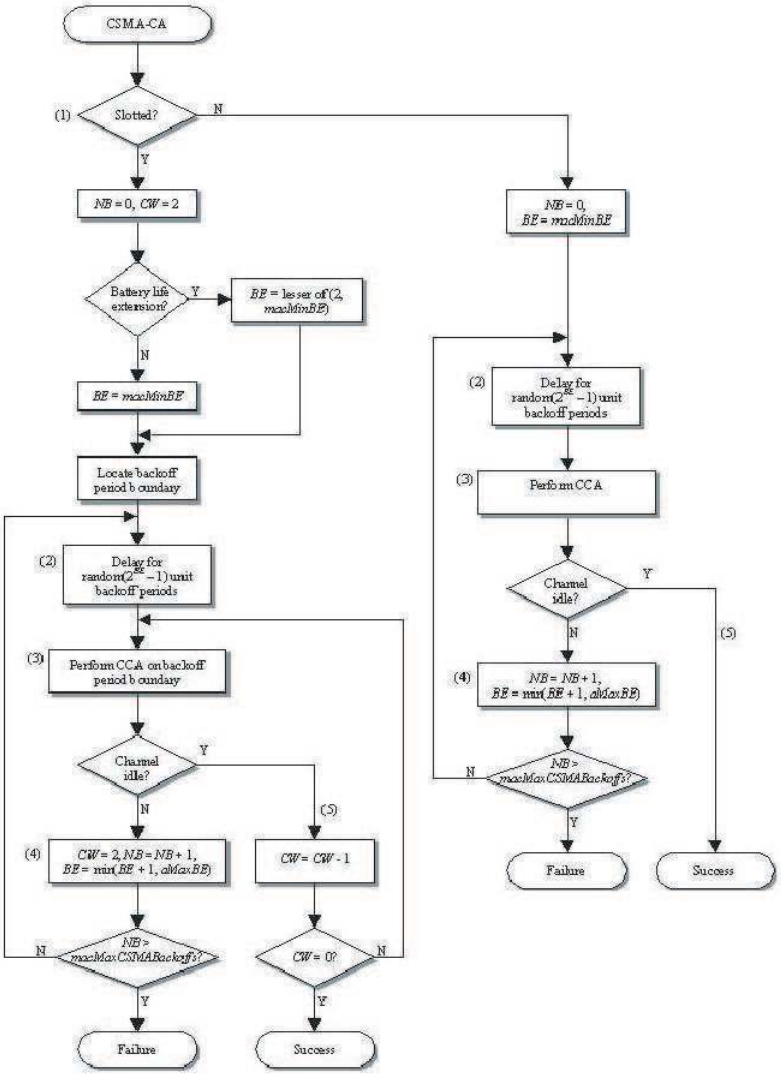

Figure 4.2: The CSMA-CA algorithm.

data frame is then sent using slotted CSMA-CA. The device acknowledged the successful reception of the data by transmitting an acknowledgement frame. Upon receiving the acknowledgement, the message is removed from the list of pending messages in the beacon as shown in Figure 4.5.

When a coordinator wishes to transfer data to a device in a nonbeacon-enabled network, it stores the data for the appropriate device to make contact and request data. A device may make contact by transmitting a MAC command requesting the data, using unslotted CSMA-CA, to its coordinator at *an application-defined rate*. The coordinator acknowledges this packet. If data are pending, the coordinator transmits the data frame using unslotted CSMA-CA. If data are not pending, the coordinator transmits a data frame with a zero-length payload to indicate that no data were pending. The device acknowledges this packet as shown in Figure 4.6.

In a peer-to-peer network, every device can communicate with any other device in its transmission radius. There are two options for this. In the first case, the node will listen constantly and transmit its data using unslotted CSMA-CA. In the second case, the nodes synchronize with each other so that they can save power.

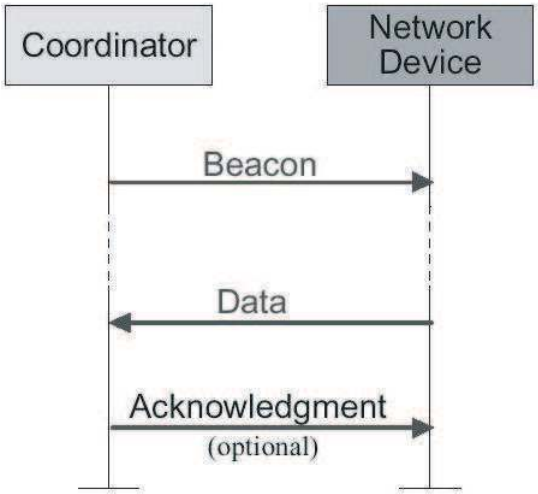

Figure 4.3: Communication to a coordinator in a beacon-enabled network.

4.4 Starting and Maintaining PANs

A PAN shall be started by an FFD only after an active channel or ED channel scan has been performed and a suitable PAN identifier selection has been made as shown in Figure 4.7. The active scan allows the FFD to locate any coordinator transmitting beacon frames within its POS (personal operating space).

An active channel scan is requested over a specified set of logical channels. For each logical channel, the device shall first switch to the channel and send a beacon request command. The device shall then enable its receiver for at most $aBaseSuperframeDuration * (2^n + 1)$ symbols, where n is between 0 and 14. During this time, the device shall reject all nonbeacon frames and record the information contained in all unique beacons in a PAN descriptor structure.

If the coordinator of a beacon-enabled PAN receives the beacon request command, it shall ignore the command and continue transmitting its beacons as usual. If the coordinator of a nonbeacon-enabled PAN receives this command, it shall transmit a single beacon frame using unslotted CSMA-CA.

The active scan on a particular channel terminates when the number of PAN descriptors stored equals this implementation-specified maximum or $aBaseSuperframeDuration * (2^n + 1)$ symbols, where n is between 0 and 14, have elapsed. The entire scan shall terminate when the number of PAN descriptors stored equals the implementation-specified maximum or every channel in the set of available channels has been scanned.

Then SELECTING a suitable PAN identifier BY prospective PAN coordinator from the list of PAN descriptors returned from the active channel scan IS UP TO APPLICATION.

An ED scan allows the FFD obtain a measure of the peak energy in each requested channel. During the ED scan, the MAC sublayer shall discard all frames received over the PHY data service. An ED scan is performed over a set of logical channels. For each logical channel, repeatedly perform an ED measurement for $aBaseSuperframeDuration * (2^n + 1)$ where n is the value of the *scanDuration*. The maximum ED measurement obtained during this period shall be noted before moving onto the next channel in the channel list. The ED scan shall terminate when either

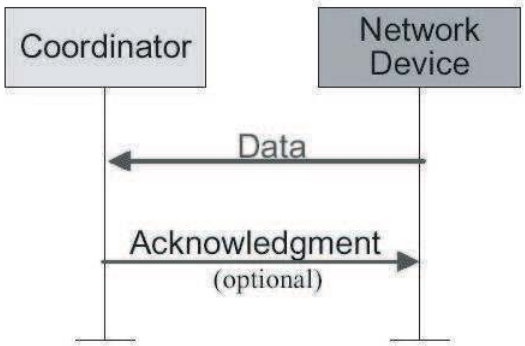

Figure 4.4: Communication to a coordinator in a nonbeacon-enabled network.

the number of channel ED measurements stored equals the implementation-specified maximum or energy has been measured on each of the specified logical channels.

In some instances, a situation could occur in which two PANs exist in the same POS with the same PAN identifier. If this conflict happens, the coordinator and its devices shall perform PAN identifier conflict resolution procedure.

The PAN coordinator shall conclude that a PAN identifier conflict is present if either a beacon frame is received by the PAN coordinator with the PAN coordinator subfield set to 1, i.e. transmitted by the PAN coordinator, and the PAN identifier is equal to macPANId or a PAN ID conflict notification command is received by the PAN coordinator from a device on its PAN. A device shall conclude that a PAN identifier conflict is present if a beacon frame is received by the device with the PAN coordinator subfield set to 1, the PAN identifier equal to *macPANId*, an address that is not equal to both *macCoordShortAddress* and *macCoordExtendedAddress*.

On the detection of the PAN identifier conflict by a device, it shall generate the PAN ID conflict notification command and send it to the PAN coordinator. If the PAN ID conflict notification command is received correctly, the PAN coordinator shall send an ack and resolve the conflict.

On the detection of the PAN identifier conflict by a coordinator, the coordinator shall first perform an active scan and then select a new PAN identifier based on the information from the scan. The coordinator shall then broadcast the coordinator realignment command containing the new PAN identifier with the source PAN identifier field equal to the value in *macPANId*. Once the coordinator realignment field has been sent, the coordinator shall set *macPANId* to the new PAN identifier.

4.5 Beacon Generation

Depending on the parameters of the MLME-START.request primitive, the FFD may either operate in a beaconless mode or may begin beacon transmissions either as the PAN coordinator or as a device on a previously established PAN. An FFD that is not the PAN coordinator shall begin transmitting beacon frames only when it has successfully associated with a PAN. This primitive also includes *macBeaconOrder* and *macSuperFrameOrder* parameters that determine the duration of the beacon interval and the duration of the active and inactive portions.

The time of the transmission of the most recent beacon shall be recorded in *macBeaconTxTime* and shall be computed so that its value is taken at the same symbol boundary in each beacon frame,

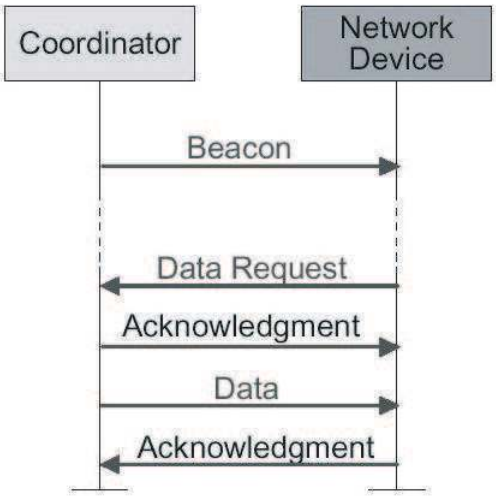

Figure 4.5: Communication from a coordinator in a beacon-enabled network.

the location of which is implementation specific.

4.6 Association and Disassociation

An FFD may indicate its presence on a PAN to other devices by transmitting beacon frames. This allows other devices to perform device discovery. An FFD that is not a PAN coordinator shall begin transmitting beacon frames only when it has successfully associated with a PAN.

Association of a device starts after having completed either an active channel scan or a passive channel scan. The passive scan, like an active scan, allows a device to locate any coordinator transmitting beacon frames within its POS whereas there beacon request command is not required for passive scan.

The results of the channel scan are then used to choose a suitable PAN. A device shall attempt to associate only with a PAN that is currently allowing association. HOW TO CHOOSE A SUITABLE PAN WITH WHICH TO ASSOCIATE FROM THE LIST OF PAN DESCRIPTORS RETURNED FROM THE CHANNEL SCAN IS UP TO APPLICATION. Following the selection of a PAN with which to associate, the next higher layers request that MLME configures the *phyCurrentChannel* to the appropriate logical channel on which to associate, *macPANId* to the identifier of the PAN with which to associate and *macCoordExtendedAddress* or *macCoordShortAddress* to the address of the coordinator with which it associates.

An unassociated device shall initiate the association procedure by sending an associate request command to the coordinator of an existing PAN. If the association request command is received correctly, the coordinator shall send an acknowledgement. This acknowledgement however does not mean that the device has associated. The coordinator needs time to determine whether the current sources available on a PAN are sufficient to allow another device to associate. This decision should be made within *aResponseWaitTime* symbols. If already associated, remove all information. If sufficient resources are available, the coordinator shall allocate a short address to the device and generate an association response command containing the new address and a status indicating

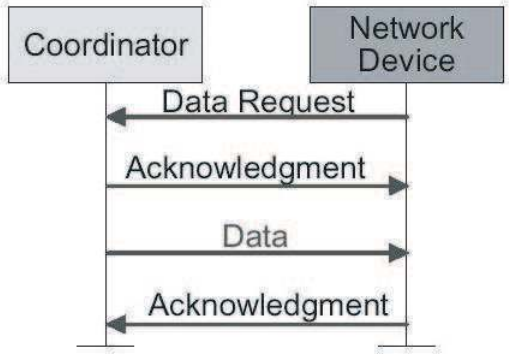

Figure 4.6: Communication from a coordinator in a nonbeacon-enabled network.

the successful association. If there are not enough resources, the coordinator shall generate an association response command containing a status indicating failure. This response is sent to the device using indirect transmission (pending, request,...).

On the other side, the device, after getting the acknowledgement frame, waits for the response for *aResponseWaitTime* symbols. It either checks the beacons in the beacon-enabled network or extract the association response command from the coordinator after *aResponseWaitTime* symbols. On reception of association response command, the device shall send an acknowledgement. If the association is successful, store the addressed of the coordinator with which it has associated.

The association procedure is shown in Figure 4.8 on the coordinator side and in Figure 4.9 on the device side.

When a coordinator wants one of its associated devices to leave the PAN, it shall send the disassociation notification command to the device using indirect transmission. Upon reception of the packet, the device should send the acknowledgement frame. Even if the ack is not received, the coordinator shall consider the device disassociated.

If an associated device wants to leave the PAN, it shall send a disassociation notification command to the coordinator. Upon reception, the coordinator sends ack. Even if the ack is not received, the device shall consider itself disassociated.

An associated device shall disassociate itself by removing all references to the PAN. A coordinator shall disassociate a device by removing all references to that device.

4.7 Synchronization

For PANs supporting beacons, synchronization is performed by receiving and decoding beacon frames. For PANs not supporting beacons, the synchronization is performed by polling the coordinator for data.

In a beacon enabled network, devices shall be permitted to acquire synchronization only with beacons containing the PAN identifier specified in *macPANId*. If tracking is specified in the MLME-SYNC.request primitive, the device shall attempt to acquire the beacon and keep track of it by regular and timely activation of its receiver. It shall enable its receiver at a time prior to the next expected beacon frame transmission, i.e. just before the known start of the next superframe. If tracking is not specified, the device shall attempt to acquire the beacon only once.

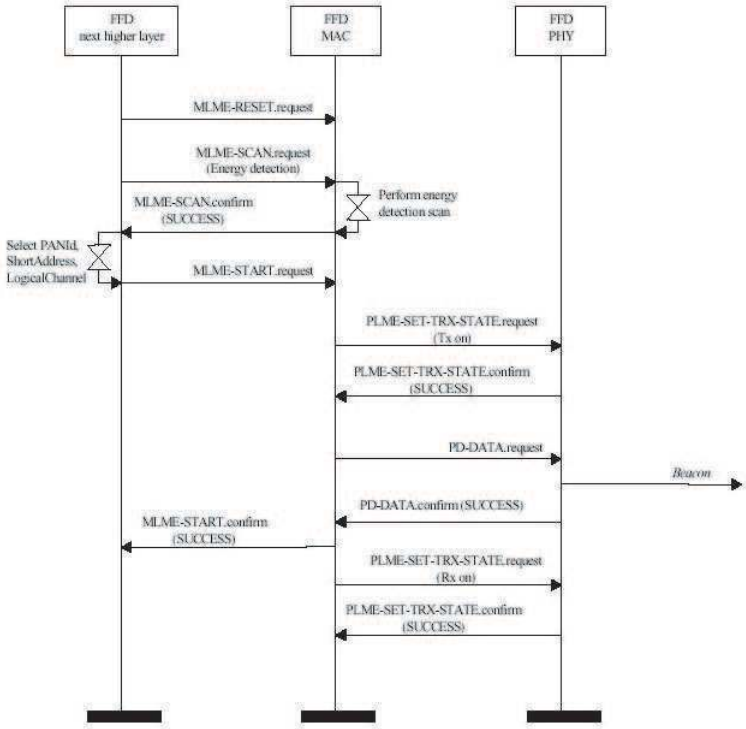

Figure 4.7: PAN start message sequence chart - PAN coordinator.

To acquire beacon synchronization, a device shall enable its receiver and search for at most $aBaseSuperframeDuration * (2^n + 1)$ symbols, where n is the *macBeaconOrder*. If a beacon frame containing the current PAN identifier of the device is not received, the MLME shall repeat the search. Once the number of missed beacons reached *aMaxLostBeacons*, the MLME notifies the next upper layer by issuing MLME-SYNC-LOSS.indication with a reason BEACON-LOSS.

The MLME shall timestamp each received beacon frame at the same symbol boundary within each frame, the location of which is implementation specific.

In a nonbeacon-enabled network, the devices shall be able to poll the coordinator for data at the discretion of the next higher layer. On receipt of MLME-POLL.request primitive, the MLME follow the procedure for extracting pending data from the coordinator.

Another problem with synchronization is **orphaned device**. If the next higher layer receives repeated communication failures following its requests to transmit data, it may conclude that it has been orphaned. A single communications failure occurs when a device transaction fails to reach the coordinator, i.e. an acknowledgement is not received after *aMaxFrameRetries* attempts at sending data. If the next higher layer concluded that the device has been orphaned, it may either reset the MAC sublayer and perform the association procedure or perform the orphaned device realignment procedure.

If the decision is for orphaned device alignment, orphan scan is performed. During the orphan scan, the MAC sublayer shall discard all frames received over the PHY data service that are not coordinator realignment MAC command frames. For each logical channel over a specified set of logical channels, the device sends an orphan notification command. The device shall then enable its receiver for at most *aResponseWaitTime* symbols. If the device successfully receives a coordinator realignment command within this time, the device shall disable its receiver.

If a coordinator receives the orphan notification command, it searches its device list for the device sending the command. If the coordinator finds a record of the device, it shall send a coordinator realignment command to the orphaned device. Otherwise, it shall ignore the packet. The orphan scan terminates when the device receives a coordinator realignment command or the specified set of logical channels has been scanned.

4.8 Transmission, Reception and Acknowledgement

In order to transmit a data or a beacon or a MAC command frame, the MAC sublayer shall copy the value of *masDSN* into the sequence number field of the MHR of the outgoing frame and then increment it by one. The source address field shall contain the address of the device sending the frame. If the device has been allocated a short address, it shall use that address in preference to its 64 bit extended address. If the source address field is not present, the originator of the frame shall be assumed to be a PAN coordinator and the destination address shall contain the address of the recipient. The destination address shall contain the intended recipient of the frame, which may be either a 16 bit short address or a 64 bit extended address. If the destination address field is not present, the recipient of the frame shall be assumed to be the PAN coordinator. The destination and source address may be in different PANs, which is identified by the PAN identifier fields.

In beacon-enabled PAN, the transmitting device shall attempt to find the beacon before transmitting. If it cannot find the beacon, it shall use unslotted CSMA-CA. Once the beacon is found, it transmits in the appropriate portion of the superframe. Transmission in the CAP shall use slotted CSMA-CA and those in GTS shall not use CSMA-CA. In a nonbeacon-enabled network, the frames are transmitted using unslotted CSMA-CA.

Upon reception of packets, the MAC sublayer shall discard all its received frames that do not contain a correct value in their FCS field in the MFR.

Receiver is important in terms of energy consumption. Each device may choose whether the MAC sublayer is to enable its receiver during idle periods. During these idle periods, the MAC sublayer shall still service transceiver task requests from the next higher layer. On completion of each transceiver task, the MAC sublayer shall request that the PHY enables or disables its receiver, depending on whether *macRxOnWhenIdle* is set to TRUE or FALSE, respectively. If beacon is enabled, the value of *macRxOnWhenIdle* shall be considered only during idle periods of the CAP.

Another energy conserving feature of the standard is the indirect transmission feature. The transactions start by the devices themselves rather than the coordinator. In other words, either the coordinator needs to indicate in its beacon when messages are pending for devices or the devices themselves need to poll the coordinator to determine whether they have any messages pending.

A device on a beacon-enabled PAN can determine whether any frames are pending for it by examining the contents of the received beacon frame. If the address of the device is contained in the address list field of the beacon frame, the MLME of the device shall send a data request command to the coordinator during the CAP. Upon reception of this command, the coordinator shall send an ack. It indicates whether any data is pending for that device in the ack frame. On receipt of the ack, the device shall enable its receiver for at most *aMaxFrameResponseTime* CAP symbols in a beacon-enabled PAN or symbols in a nonbeacon-enabled PAN to receive the corresponding frame from the coordinator. If there is data pending, the coordinator should send the frame else send a frame containing zero length payload, indicating that no data is present.

The data frame is transmitted without using CSMA-CA if the MAC sublayer can commence transmission of the data frame between $aTurnaroundTime$ and $aTurnaroundTime+aUnitBackoffPeriod$ symbols and there is time remaining in the CAP for the message, appropriate IFS and acknowledge-

ment and using CSMA-CA otherwise.

A frame transmitted with the acknowledgement request field set to 1 shall be acknowledged by the recipient. If the intended recipient correctly receives the frame, it shall generate and send an acknowledgement frame containing the same DSN from the data or MAC command frame that is being acknowledged. The transmission of the ack shall commence between $aTurnaroundTime$ and $aTurnaroundTime + aUnitBackoffPeriod$ symbols after the reception of the last symbol of the data or MAC command frame.

4.9 GTS Allocation and Management

A GTS allows a device to operate on the channel within a portion of the superframe that is dedicated exclusively to that device. A device shall attempt to allocate and use a GTS only if it is currently tracking the beacons. A GTS shall be allocated only by the PAN coordinator and it shall be used only for communications between the PAN coordinator and a device. A single GTS can extend over one or more superframe slots. The PAN coordinator may allocate up to seven GTSs at the same time, provided there is sufficient capacity in the superframe.

A GTS shall be allocated before use, with the PAN coordinator deciding whether to allocate a GTS based on the requirements of the GTS request and the current available capacity in the superframe. GTS shall be allocated on a first-come-first-serve basis and all GTSs shall be placed contiguously at the end of the superframe and after the CAP. Each GTS shall be deallocated when the GTS is no longer required, and a GTS can be deallocated at any time at the discretion of the PAN coordinator or by the device that originally requested the GTSs. A device that has been allocated GTS may also operate in the CAP.

The management of the GTSs shall be undertaken by the PAN coordinator only. For each GTS, the PAN coordinator shall be able to store its starting slot, length, direction and associated device address.

The GTS direction is specified as either transmit or receive. Each device may request one transmit GTS and/or one receive GTS. For each allocated GTS, the device shall be able to store its starting slot, length and direction. If a device has been allocated a receive GTS, it shall enable its receiver for the entirety of the GTS. In the same way, a PAN coordinator shall enable its receiver for the entirety of the GTS if a device has been allocated a transmit GTS.

A device is instructed to request the allocation of a new GTS through the GTS request command, with GTS characteristics (direction, length,...) set according to the requirements of the intended application. On receipt of this command, the PAN coordinator shall send an acknowledgement frame. Following the ack transmission, the PAN coordinator shall first check if there is available capacity in the current superframe based on the remaining length of the CAP and the desired length of the requested GTS. The superframe shall have available capacity if the maximum number of GTSs has not been reached and allocating a GTS of the desired length would not reduce the length of the CAP to less than *aMinCAPLength*. The PAN coordinator shall make its decision within *aGTSDescPersistenceTime* superframes. On receipt of the ack from the coordinator, the device shall continue to track the beacons and wait for at most *aGTSDescPersistenceTime* superframes. If no GTS decsriptor in the superframe, notify the next upper layer of failure.

When the coordinator determines whether capacity is available for the requested GTS, it shall generate a GTS descriptor with the requested specifications and the short address of the requested device. It indicates the length and the start of the GTS in the superframe and notifies the next upper layer of the new GTS allocation. If there was not sufficient capacity to allocate the requested GTS, the start slot shall be set to 0 and the length to the largest GTS length that can currently be supported.

This GTS descriptor shall remain in the beacon frame for *aGTSPersistenceTime* superframes.

On receipt of the beacon frame, the device shall process the descriptor and notify the next upper layer of the success.

In the same way, a device is instructed to request the deallocation of an existing GTS through the GTS request command using the characteristics of the GTS it wishes to deallocate. From this point on, the GTS to be deallocated shall not be used by the device. Then an ack from the PAN coordinator to the device. The PAN coordinator then deallocates the request of the GTS characteristics in the packet matches those in its allocation. The PAN coordinator shall also ensure that any gaps occurring in the CFP, appearing due to the deallocation of a GTS, are removed to maximize the length of the CAP.

The MLME of the PAN coordinator shall also attempt to detect when a device has stopped using a GTS using the following rules: For a transmit frame GTS, the MLME of the PAN coordinator shall assume that the device is no longer using the GTS if a data frame is not received for at least $2 * n$ superframes. For receive GTSs, the MLME of the PAN coordinator shall assume that the device is no longer using its GTS if an acknowledgement frame is not received for at least $2 * n$ superframes. The value of n is equal to $2^{8-macBeaconOrder}$ if $0 \leq macBeaconOrder \leq 8$ and 1 if $9 \leq macBeaconOrder \leq 14$.

4.10 MAC Frame Formats

The general MAC frame format is given in Figure 4.10.Each MAC frame consists of the following basic components:

- MHR, which comprises frame control, sequence number, and address information
- A MAC payload of variable length, which contains information specific to the frame type. Acknowledgement frames do not contain a payload.
- A MFR, which contains FCS.

LR-WPAN defines 4 frame structures: beacon frame (Figure 4.11), data frame (Figure 4.12), acknowledgement frame (Figure 4.13), MAC command frame (Figure 4.14).

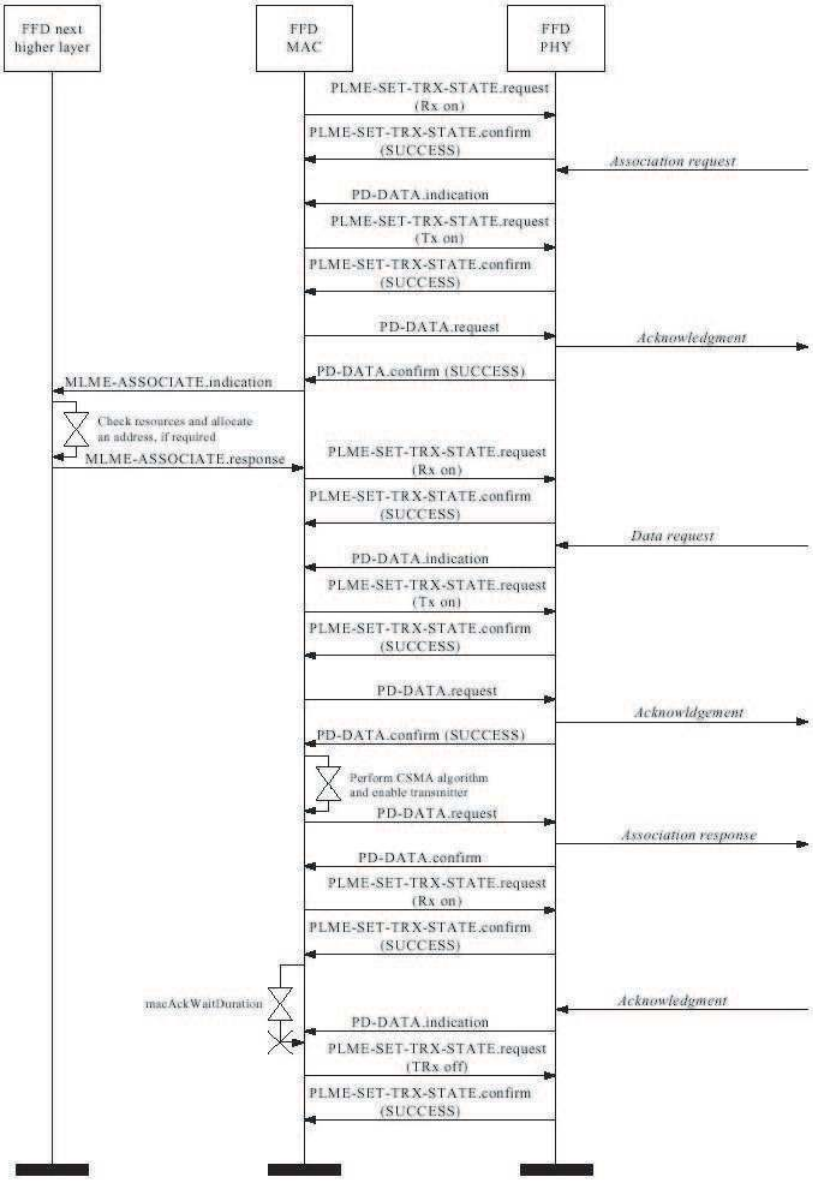

Figure 4.8: Association message sequence chart - coordinator.

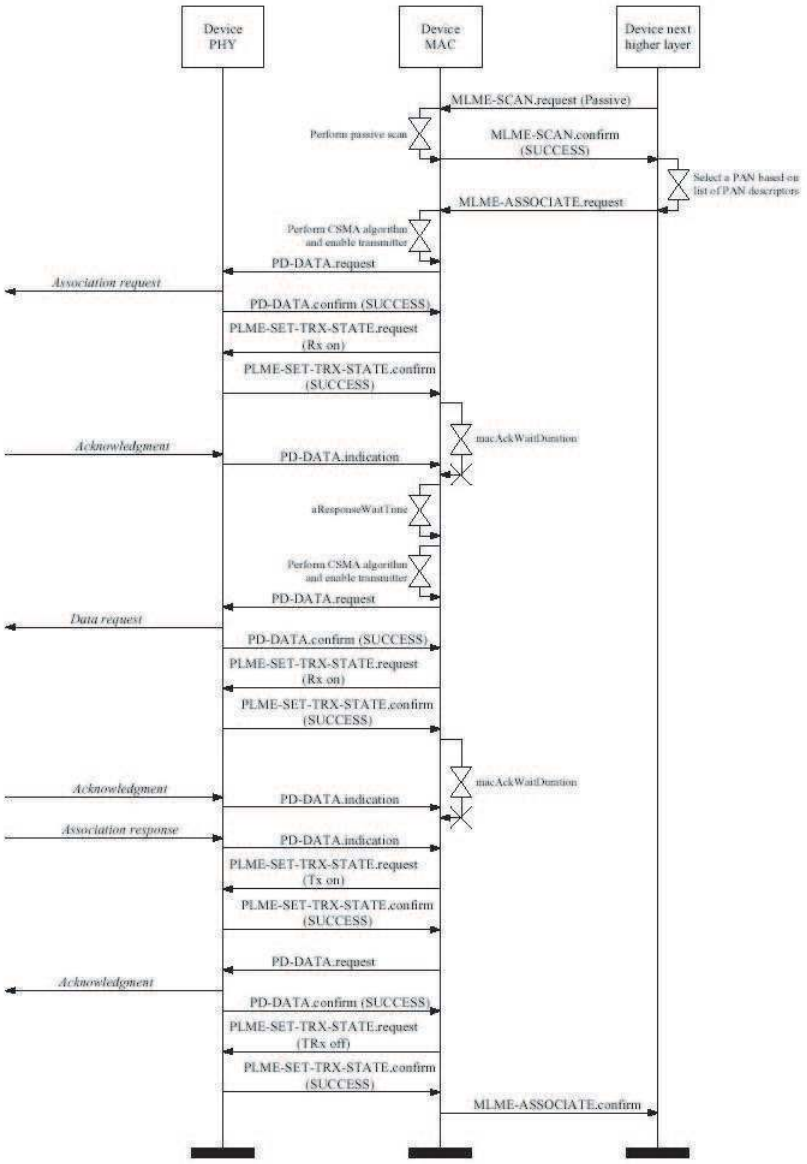

Figure 4.9: Association message sequence chart - device.

Octets: 2	**1**	**0/2**	**0/2/8**	**0/2**	**0/2/8**	**variable**	**2**
Frame control	Sequence number	Destination PAN identifier	Destination address	Source PAN identifier	Source address	Frame payload	FCS
		Addressing fields					
MHR						MAC payload	MFR

Figure 4.10: General MAC frame format.

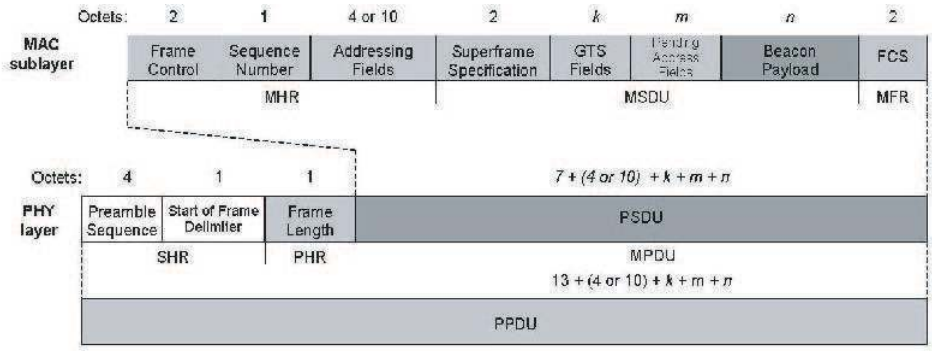

Figure 4.11: Schematic view of the beacon frame.

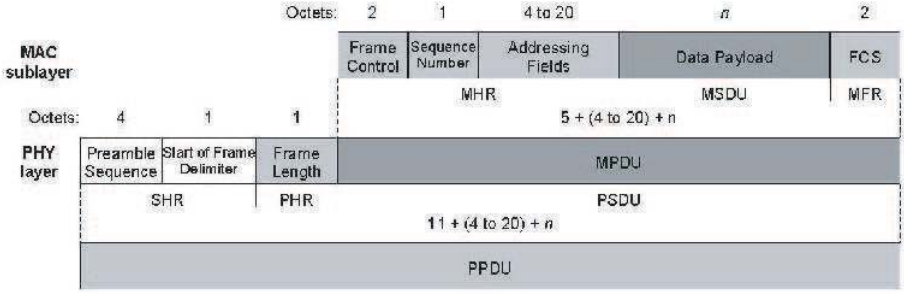

Figure 4.12: Schematic view of the data frame.

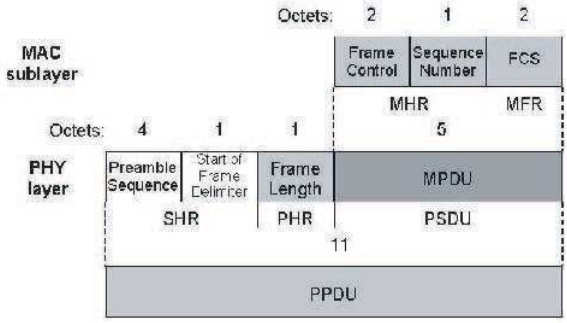

Figure 4.13: Schematic view of the acknowledgement frame.

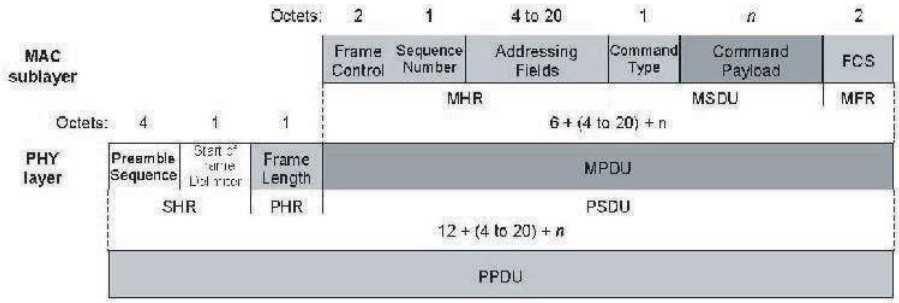

Figure 4.14: Schematic view of the MAC command frame.

Chapter 5

ZigBee Routing Layer

ZigBee routing algorithm can be thought of an hierarchical routing strategy with table-driven optimizations applied where possible. The routing layer is said to start with the well-studied public-domain algorithm Ad hoc On Demand Distance Vector (AODV) and Motorola's Cluster-Tree algorithm.

We summarize AODV in Section 5.1 and Cluster-Tree in Section **??**.

5.1 AODV: Ad hoc On Demand Distance Vector

AODV is a pure on-demand route acquisition algorithm: nodes that do not lie on active paths neither maintain any routing information nor participate in any periodic routing table exchanges. Further, a node does not have to discover and maintain a route to another node until the two need to communicate, unless the former node is offering services as an intermediate forwarding station to maintain connectivity between two other nodes.

The primary objectives of the algorithm are to broadcast discovery packets only when necessary, to distinguish between local connectivity management and general topology maintenance and to disseminate information about changes in local connectivity to those neighboring mobile nodes that are likely to need the information.

When a source node needs to communicate with another node for which it has no routing information in its table, the *Path Discovery* process is initiated. Every node maintains two separate counters: *sequence number* and *broadcast id*. The source node initiates path discovery by broadcasting a route request (RREQ) packet to its neighbors, which includes *source addr, source sequence number, broadcast id, dest addr, dest sequence number, hop cnt*. (Source sequence number is for maintaining freshness information about the reverse route whereas the destination sequence number is for maintaining freshness of the route to the destination before it can be accepted by the source.)

The pair *source addr, broadcast id* uniquely identifies a RREQ, where *broadcast id* is incremented whenever the source issues a new RREQ. When an intermediate node receives a RREQ, if it has already received a RREQ with the same *broadcast id* and source address, it drops the redundant RREQ and does not rebroadcast it. Otherwise, it rebroadcasts it to its own neighbors after increasing *hop cnt*. Each node keeps the following information: destination IP address, source IP address, broadcast id, expiration time for reverse path route entry and source node's sequence number.

As the RREQ travels from a source to destinations, it automatically sets up the *reverse path* from all nodes back to the source. To set up a reverse path, a node records the address of the neighbor from which it received the first copy of RREQ. These reverse path route entries are maintained for at least enough time for the RREQ to traverse the network and produce a reply to the sender.

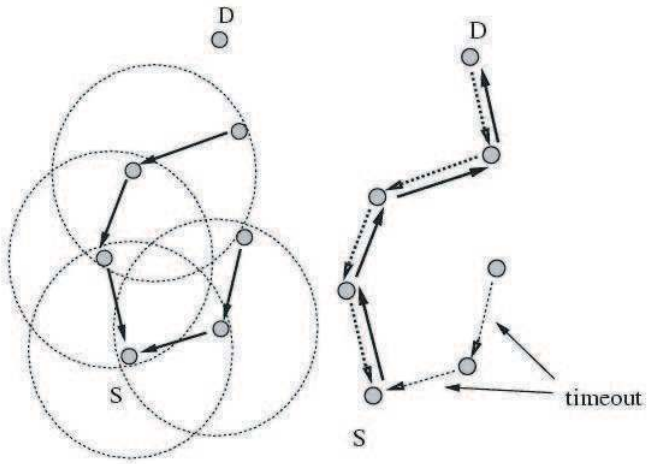

Figure 5.1: Reverse and forward path formation in AODV protocol.

When the RREQ arrives at a node, possibly the destination itself, that possesses a current route to the destination, the receiving node first checks that the RREQ was received over a bi-directional link. If this node is not destination but has route to the destination, it determines whether the route is current by comparing the destination sequence number in its own route entry to the destination sequence number in the RREQ. If RREQ's sequence number for the destination is greater than that recorded by the intermediate node, the intermediate node must not use this route to respond to the RREQ, instead rebroadcasts the RREQ.

If the route has a destination sequence number that is greater than that contained in the RREQ or equal to that contained in the RREQ but a smaller hop count, it can unicasts a route reply packet (RREP) back to its neighbor from which it received the RREQ. A RREP contains the following information: source addr, dest addr, dest sequence number, hop cnt and lifetime. As the RREP travels back to the source, each node along the path sets up a forward pointer to the node from which the RREP came, updates its timeout information for route entries to the source and destination, and records the latest destination sequence number for the requested destination.

Nodes that are along the path determined by the RREP will timeout after route request expiration timer and will delete the reverse pointers since they are not on the path from source to destination as shown in Figure 5.1. The value of this timeout time depends on the size of the ad hoc network. Also, there is the routing caching timeout that is associated with each routing entry to show the time after which the route is considered to be invalid. Each time a route entry is used to transmit data from a source toward a destination, the timeout for the entry is reset to the current time plus active-route-timeout.

The source node can begin data transmission as soon as the first RREP is received, and can later update its routing information if it learns of a better route.

Each routing table entry includes the following fields: destination, next hop, number of hops (metric), sequence number for the destination, active neighbors for this route, expiration time for the route table entry.

For path maintenance, each node keeps the address of active neighbors through which packets for the given destination are received is maintained. This neighbor is considered active if it originates or relays at least one packet for that destination within the last active-timeout period. Once the

next hop on the path from source to the destination becomes unreachable (hello messages are not received for a certain time, hello messages also ensures that only nodes with bidirectional connectivity are considered to be neighbors, therefore each hello message included the nodes from which the node has heard), the node upstream of the break propagates an unsolicited RREP with a fresh sequence number and hop count of ∞ to all active upstream nodes. This process continues until all active source nodes are notified. Upon receiving the notification of a broken link, the source nodes can restart the discovery process if they still require a route to the destination. If it decides that it would like to rebuild the route to the destination, it sends out an RREQ with a destination sequence number of one greater than the previously known sequence number, to ensure that it builds a new, viable route and that no nodes reply if they still regard the previous route as valid.

5.2 Cluster-Tree Algorithm

The cluster-tree protocol is a protocol of the logical link and network layers that uses link-state packets to form either a single cluster network or a potentially larger cluster tree network. The network is basically self-organized and supports network redundancy to attain a degree of fault resistance and self-repair.

Nodes select a cluster head and form a cluster according to the self-organized manner. Then self-developed clusters connect to each other using the Designated Device (DD).

5.2.1 Single Cluster Network

The cluster formation process begins with cluster head selection. After a cluster head is selected , the cluster head expands links with other member nodes to form a cluster.

After a node turns on, it scans the channels to search for a HELLO message form other nodes (HELLO messages correspond to beacons in MAC layer of IEEE 802.15.4). If it can't get any HELLO messages for a certain time, then it turns to a cluster head as shown in Figure 5.2 and sends out HELLO messages to its neighbors. The new cluster head wait for responses from neighbors for a while. If it hasn't received any connection requests, it turns back to a regular node and listens again. The cluster head can also be selected based on stored parameters of each node, like transmission range, power capacity, computing ability or location information.

After becoming the cluster head (CH), the node broadcasts a periodic HELLO message that contains a part of the cluster head MAC address and node ID 0 that indicates the cluster head. The nodes that receive this message send a CONNECTION REQUEST message to the cluster head. When the CH receives it, it responds to the node with a CONNECTION RESPONSE message that contains a node ID for the node (node ID corresponds to the short address at the MAC layer). The node that is assigned a node ID replies with an ACK message to the cluster head. The message exchange is shown in Figure 5.3.

If all nodes are located in the range of the cluster head, the topology of connection becomes a star and every member nodes are connected to the cluster head with one hop. A cluster can expand into a multi-hop structure when each node supports multiple connections. The message exchange for the multihop cluster set up procedure is shown in Figure 5.4.

If the cluster head has run out of all node IDs or the cluster has reached some other defined limit, it should reject connection requests from new nodes. The rejection is through the assignment of a special ID to the node.

The entry of the neighbor list and the routes is updated by the periodic HELLO message. If a node entry does not update until a certain timeout limit, it should be eliminated.

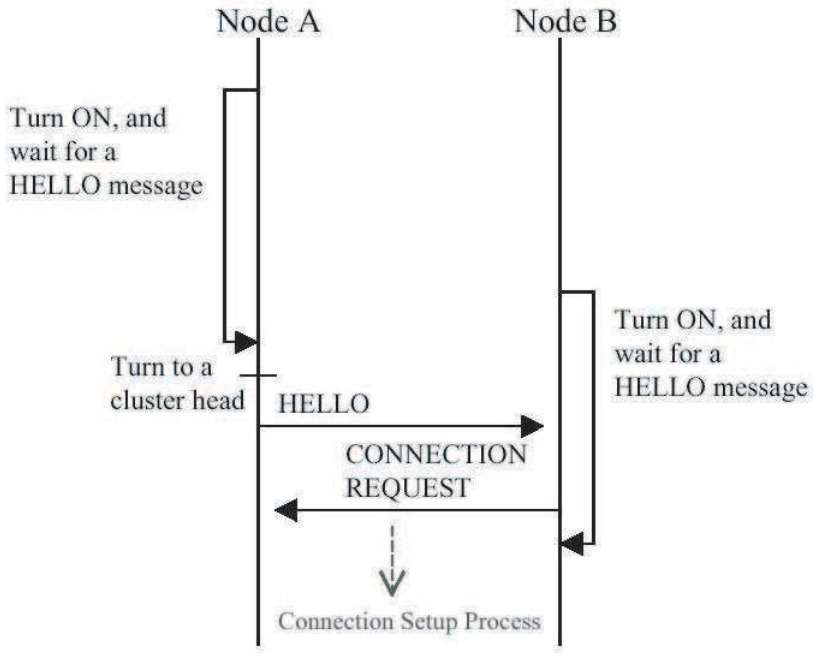

Figure 5.2: Cluster head selection process.

A node may receive a HELLO message from a node that belongs to different cluster. In that case, the node adds the cluster ID (CID) of the transmitting node in the neighbor list and then sends it inside a LINK STATE REPORT to the CH so that CH knows which clusters its cluster has intersection.

The LINK STATE REPORT message also contain the neighbors node ID list of the node so that the CH knows the complete topology to make topology optimizations. If the topology change is required, then the CH sends a TOPOLOGY UPDATE message. If a member receives a TOPOLOGY UPDATE message that the different parent node is linked to the node, it changes the parent node as indicated in the message. And it also records its child nodes and the nodes below it in the tree at this time.

If a member node has trouble and becomes unable to communicate, the tree route of the cluster would be reconfigured. The CH know the presence of a trouble by the periodic LINK STATE REPORT. When the cluster head has trouble, the distribution of HELLO message is stopped and all member nodes know that they have lost the CH. The cluster would then be reconfigured in the same way as the cluster formation process.

5.2.2 Multi-Cluster Network

To form a network, a Designated Device (DD) is needed. The DD has responsibility to assign a unique cluster ID to each cluster head. This cluster ID combined with the node ID that the CH assigns to each node within a cluster forms a logical address and is used to route packets. Another role of the DD is to calculate the shortest route from the cluster to the DD and inform it to all nodes within the network.

When the DD joins the network, it acts as the CH of cluster 0 and starts to send HELLO message to the neighborhood. If a CH has received this message, it sends a CONNECTION REQUEST message and joins the cluster 0. After that, the CH requests a CID to the DD. In this case, the CH is

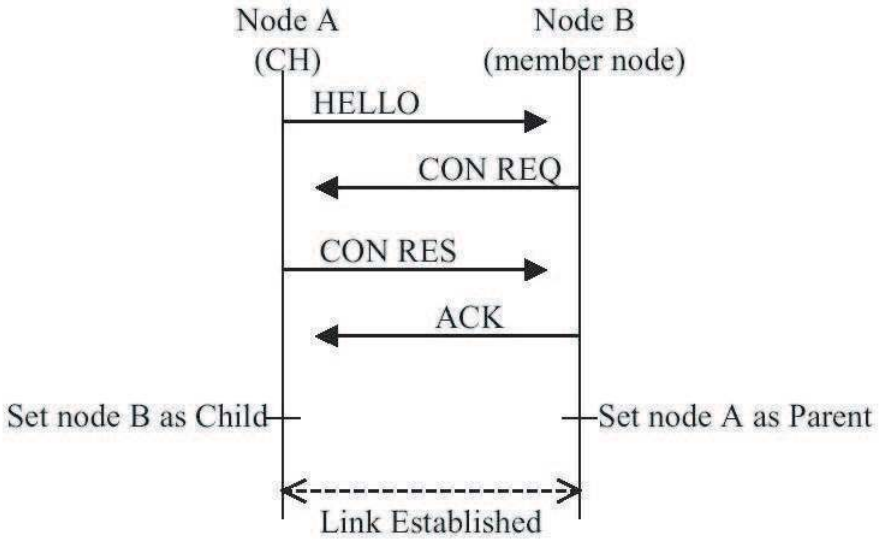

Figure 5.3: Link setup between CH and member node.

a border node that has two logical addresses. One is for a member of the cluster 0 and the other is for a CH. When the CH gets a new CID, it informs its member nodes by the HELLO message.

If a member has received the HELLO message from the DD, it adds CID 0 in its neighbor list and reports to its CH. The reported CH selects the member node as a border node to its parent cluster and sends a NETWORK CONNECTION REQUEST message to the member node to set up a connection with the DD. The border node requests a connection and joins the cluster 0 as its member node. Then it sends a CID REQUEST message to the DD. After the CID RESPONSE message arrival, the border node sends NETWORK CONNECTION RESPONSE message that contains a new CID to the CH. When the CH gets a new CID, it informs to its member nodes by the HELLO message.

The clusters not bordering cluster 0 use intermediate clusters to get a CID. Again, either the CH becomes the border node to its parent cluster or the CH names a member node as the border to its parent cluster. These processes are shown in Figures 5.5,5.6,5.7,5.8.

Each member node of the cluster has to record its parent cluster, child/lower clusters and the border node IDs associated with both the parent and child clusters. The DD should store the whole tree structure of the clusters.

Like the nodes in the clusters, the CHs report their link state information to the DD. The CH periodically sends a NETWORK LINK STATE REPORT message that contains its neighbor cluster CID list to the DD. Then this information can be used to calculate the optimized route and periodically update the topology for the network redundancy. In the same way, the DD can send TOPOLOGY UPDATE message to inform up-to-date route from the DD to the clusters.

A backup DD (BDD) can be prepared to prevent network down time due to the DD trouble.

Inter-cluster communication, which is shown in Figure 5.9, is realized by routing. The border nodes act as routers that connect clusters and relay packets between the clusters. When a border node receives a packet, it examines the destination address, then forwards to the next border node in the adjacent cluster or to the destination node within the cluster.

Only the DD can send a message to all the nodes within its network. The message is forwarded along the tree route of clusters. The border node should forward the broadcast packet from the

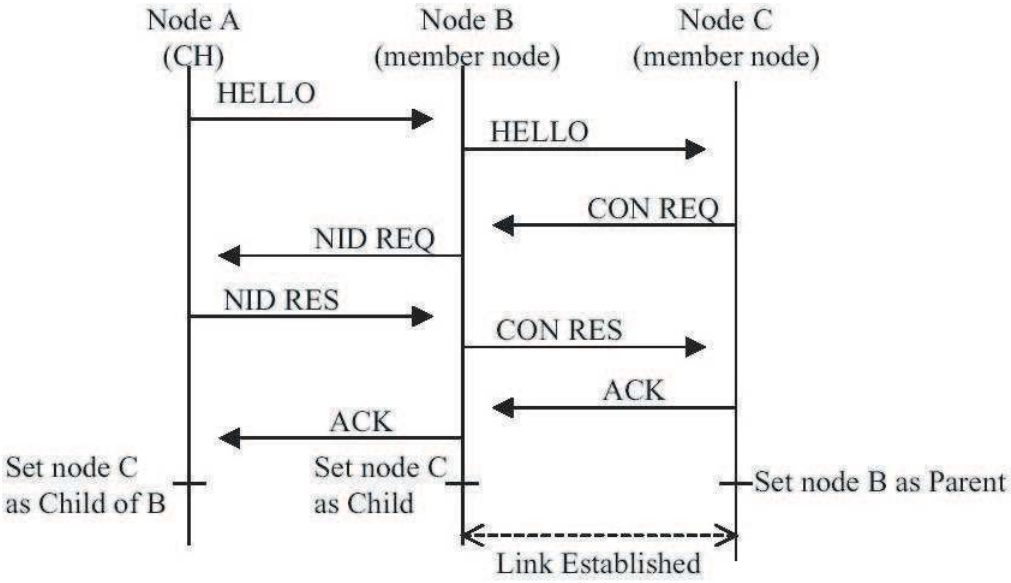

Figure 5.4: Multi hop cluster setup procedure.

parent cluster to the child cluster.

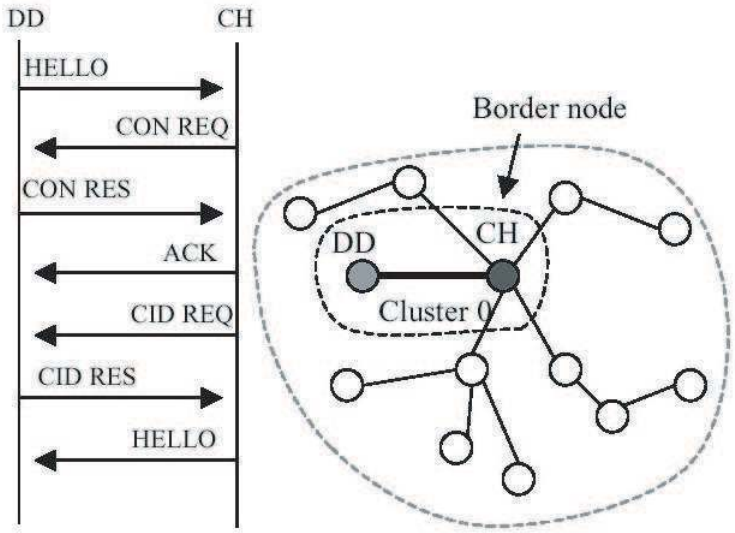

Figure 5.5: CID assignment 1.

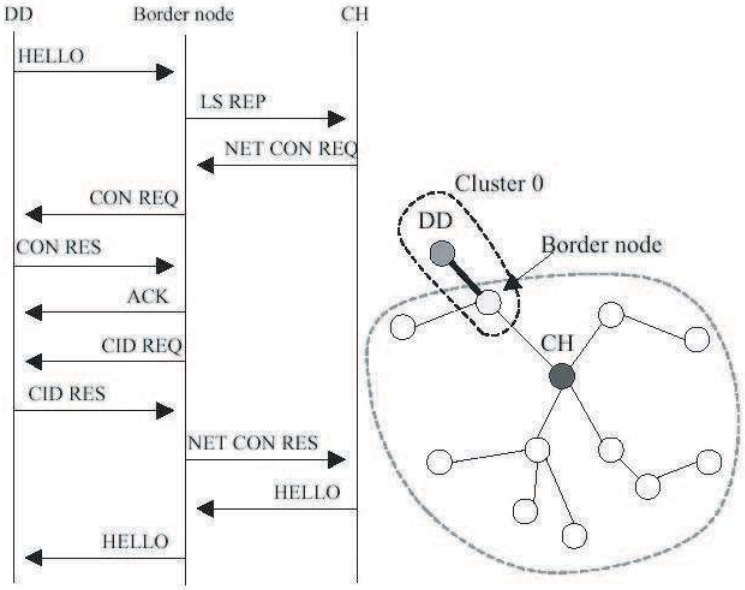

Figure 5.6: CID assignment 2.

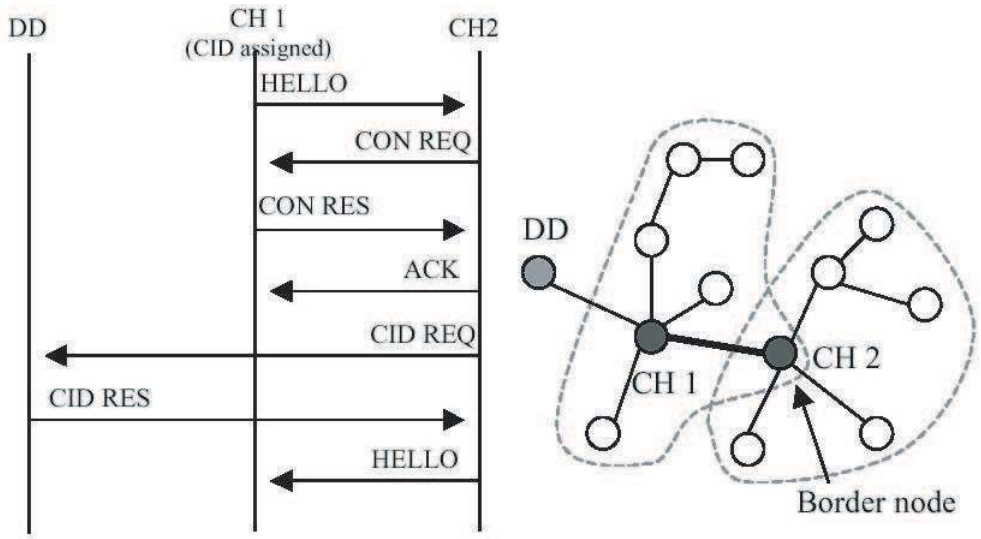

Figure 5.7: CID assignment 3.

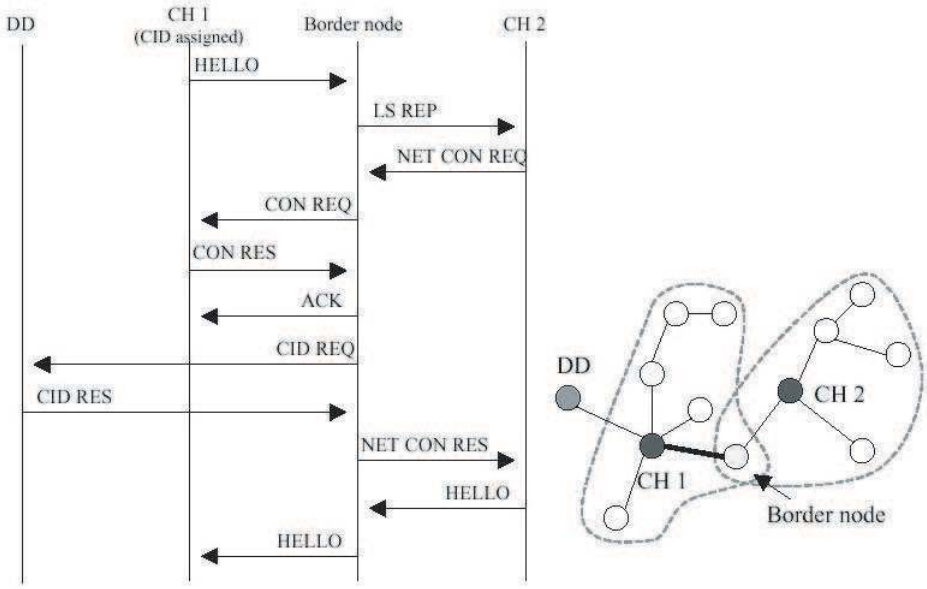

Figure 5.8: CID assignment 4.

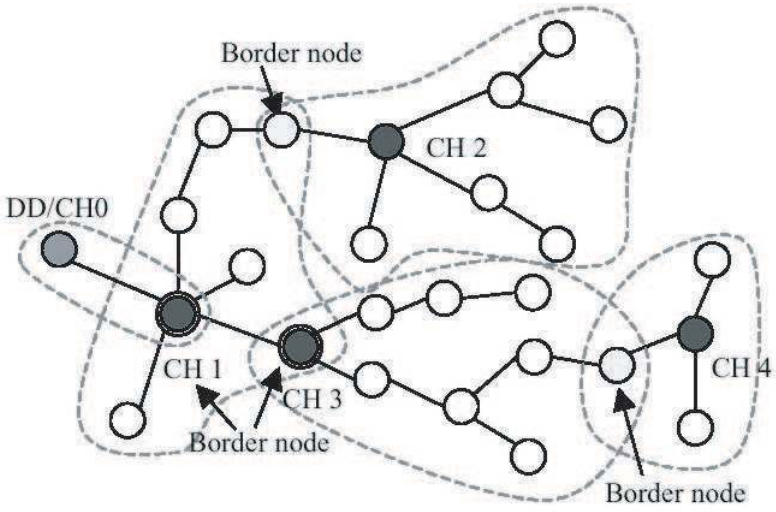

Figure 5.9: A multi cluster network and the border nodes.

Bibliography

[1] ZigBee Alliance, *http://www.caba.org/standard/zigbee.html*.

[2] LAN-MAN Standards Committee of the IEEE Computer Society, *Wireless LAN medium access control(MAC) and physical layer(PHY) specification*, IEEE, New York, NY, USA, IEEE Std 802.11-1997 edition, 1997

[3] LAN-MAN Standards Committee of the IEEE Computer Society, *Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)*, IEEE, 2003

[4] C. E. Perkins and E. M. Royer, *Ad Hoc On Demand Distance Vector Routing*

[5] IEEE P802.15 Working Group for WPANs, *Cluster Tree Network*, April 2001